MONTE CARLO BAYESIAN FILTERING AND SMOOTHING FOR TVAR SIGNALS IN SYMMETRIC α-STABLE NOISE

Marco J. Lombardi[*] *and Simon J. Godsill*

Dipartimento di Statistica "G. Parenti"
Università degli studi di Firenze
Viale G.B. Morgagni 59
I-50134 Firenze, Italy (Europe)
phone: +39 055 4237256,
fax: +39 055 4223560,
email: mjl@ds.unifi.it
web: www.ds.unifi.it/˜mjl/

Signal Processing Laboratory
Department of Engineering
University of Cambridge
Trumpington Street
CB2 1PZ Cambridge, U.K. (Europe)
phone: +44 1223 332604,
fax: +44 1223 332662,
email: sjg@eng.cam.ac.uk
web: www-sigproc.eng.cam.ac.uk/˜sjg/

ABSTRACT

In this paper we propose an on-line Bayesian filtering and smoothing method for time series models with heavy-tailed alpha-stable noise, with a particular focus on TVAR models. We first point out how a filter that fails to take into account the heavy-tailed character of the noise performs poorly and then examine how an α-stable based particle filter can be devised to overcome this problem. The filtering methodology is based on a scale mixtures of normals (SMiN) representation of the α-stable distribution, which allows efficient Rao-Blackwellised implementation within a conditionally Gaussian framework, and requires no direct evaluation of the α-stable density, which is in general unavailable in closed form. The methodology is shown to work well, outperforming the traditional Gaussian methods both on simulated and real audio data. The analysis of real degraded audio samples highlights the fact that α-stable distributions are particularly well suited for noise modelling in a realistic scenario.

1. INTRODUCTION

In many real signal processing environments, sources of noise cannot be considered as Gaussian. Here we consider in particular the estimation of processes observed in heavy-tailed noise, i.e. noise having occasional very large values that could strongly affect any inference procedure for the underlying process. We focus on one particular class of heavy-tailed distribution, the *symmetric α-stable* class, that can be obtained through a generalisation of the central limit theorem when the random variables need not have finite variance, see [7, 22]. Such noise processes have been found appropriate in a number of areas, including signal processing [20] and econometrics [23].

We propose methodology for optimal on-line estimation of stochastic processes observed in α-stable noise. The models chosen are time varying autoregressions (TVAR), which are appropriate for a wide range of signals, including speech, audio, eeg and seismic data. On-line estimation and signal extraction is performed by means of sequential Monte Carlo methods [11, 13, 4]. These methods have been applied to the TVAR model with Gaussian noise in [6, 24, 10]. This paper constitutes an extension to these approaches (and especially that of [24]) to the symmetric α-stable case and, more generally, to any case where the noise distribution can be represented as a scale mixture of normals (SMiN). The SMiN representation of the α-stable class allows us importantly to employ conditionally linear and Gaussian computational methods within the Monte Carlo filter (via the Kalman filter), hence avoiding any direct evaluations of the noise density function (which is unavailable in most cases for the α-stable model). In [9](see also further references on `www-sigproc.eng.cam.ac.uk/sjg`) methodology was presented for inference about static AR models in stable law noise. Here this methodology is extended to the sequential Monte Carlo setting for TVAR models in α-stable noise. The methods are able accurately to reconstruct the signal process, the TVAR parameters, and also the stable law parameter α, which is static and thus not easily amenable to particle filter analysis. A real application of the methods is presented for audio signal enhancement. We present compelling experimental evidence that the α-stable distribution is appropriate for certain noise sources in 78rpm gramophone disk recordings which are typically degraded by non-Gaussian clicks. Results are found to be very effective.

The structure of the paper is as follows: we begin describing the properties of α-stable distributions and proposing that they are more appropriate than other parametric families in modelling noise in certain audio sources. We then introduce the statistical model and its state-space representation. Bayesian methods are presented for sequential estimation and filtering and we discuss how symmetric α-stable distributions can be embedded into this framework. The approach is then compared to the traditional Gaussian framework on both simulated data and artificially corrupted audio samples. An application to real audio data concludes the paper.

2. α-STABLE DISTRIBUTIONS

The α-stable family of distributions stems from a more general version of the "traditional" central limit theorem in which the assumption of finite variance is replaced by a much less restrictive one concerning the regular behavior of the tails [7]; the Gaussian distribution then becomes a particular case of α-stable distribution. This family of distributions has a very interesting pattern of shapes, allowing for asymmetry and thick tails, that makes them suitable for the modelling of several phenomena; moreover, it is closed under linear combinations.

The family is identified by means of the characteristic

[*]This paper was completed when the first author was visitng the Signal Processing lab at the University of Cambridge. We are indebted to Jaco Vermaak for allowing us to exploit and modify his Matlab source code.

function

$$\varphi(t) = \begin{cases} \exp\left\{i\delta t - \gamma^\alpha |t|^\alpha \left[1 - i\beta \operatorname{sgn}(t) \tan \frac{\pi\alpha}{2}\right]\right\} & \alpha \neq 1 \\ \exp\left\{i\delta t - \gamma|t| \left[1 + i\beta \frac{2}{\pi} \operatorname{sgn}(t) \ln |t|\right]\right\} & \alpha = 1 \end{cases} \tag{1}$$

which depends on four parameters: $\alpha \in (0, 2]$, measuring the tail thickness (thicker tails for smaller values of the parameter), $\beta \in [-1, 1]$ determining the degree and sign of asymmetry, $\gamma > 0$ (scale) and $\delta \in \mathbb{R}$ (location). To denote a stable distribution with parameters α, β, γ and δ we will use the shorthand notation $\mathcal{S}(\alpha, \beta, \gamma, \delta)$. As in the Gaussian case, a random variable X with $\mathcal{S}(\alpha, \beta, \gamma, \delta)$ distribution can be standardized to produce

$$Z = \frac{X - \delta}{\gamma} \sim \mathcal{S}(\alpha, \beta, 1, 0).$$

For the standardized stable distribution, we will henceforth use the shorthand notation $\mathcal{S}(\alpha, \beta)$.

Unfortunately, (1) can be inverted to yield a closed-form density function only for a very few cases: $\alpha = 2$, corresponding to the normal distribution, $\alpha = 1$ and $\beta = 0$, yielding the Cauchy distribution, and $\alpha = \frac{1}{2}, \beta = 1$ for the Lévy distribution. This difficulty, coupled with the fact that moments of order greater than α do not exist whenever $\alpha \neq 2$, has made impossible the use of "traditional" estimation methods such as maximum likelihood and the method of moments. Researchers have thus devised alternative estimation methods, mainly based on quantiles [17], the performance of which is judged unsatisfactory in a number of respects, especially because they are not liable to be incorporated in complex models and thus require a two-step estimation approach. With the availability of powerful computing machines, it has become possible to devise computationally-intensive estimation methods for the estimation of α-stable distributions, such as maximum likelihood based on the FFT of the characteristic function, as in [18], or direct numerical integration as in [21]. These methods, however, present some inconvenience: the accuracy of both the FFT and the numerical integration of the characteristic function is quite poor for small values of α because of the spikiness of the density function; furthermore, when the parameters are near their boundary, their distributions assume non-standard form, making traditional confidence intervals unreliable.

Given these computational difficulties, it is perhaps surprising that simulated values from α-stable distributions can be straightforwardly produced with a simple analytic transformation of two uniformly distributed random numbers [3]. The possibility of a simulation based Bayesian approach was first put forth in [2], who shows how to devise an auxiliary variable conditional on which the likelihood can be expressed in closed form. Unfortunately, simulated values from this auxiliary variable cannot readily be generated and one must resort to more elaborate sampling methods. Furthermore, several reparameterizations are needed in order to obtain posterior distributions that can be easily simulated from. This makes the whole procedure quite slow, especially when large sample sizes are involved.

In the case of symmetric stable distributions, the situation is much less cumbersome: we can exploit the fact that a symmetric α-stable distribution can be represented as a scale mixture of normals (SMiN) [1]; this observation will enable the efficient conditionally Gaussian simulation techniques proposed later. To put it in a more formal way, let us consider a generic model with symmetric α-stable noise

$$\epsilon_i \sim \mathcal{S}(\alpha, 0, \gamma, \delta).$$

If we introduce an auxiliary white noise $u_i \sim \mathcal{N}(0, 1)$, where $\mathcal{N}(0, 1)$ denotes a standard normal distribution, the above SMiN property allows us to express the α-stable noise equivalently as

$$\epsilon_i = \delta + \gamma\sqrt{\lambda_i} u_i, \qquad \lambda_i \sim \mathcal{S}\left(\tfrac{\alpha}{2}, 1\right), \; u_i \sim \mathcal{N}(0, 1).$$

where λ_i and its positive stable distribution $\mathcal{S}\left(\frac{\alpha}{2}, 1\right)$ are known as the *mixing parameter* and the *mixing distribution*, respectively.

Conditionally on λ_i, we have thus

$$\epsilon_i | \lambda_i \sim \mathcal{N}(\delta, \gamma^2 \lambda_i). \tag{2}$$

and hence in a Monte Carlo environment we will be able to utilise efficient normal/linear computations while avoiding entirely the need to evaluate $\mathcal{S}(\alpha, 0, \gamma, \delta)$. It is worth noting that this SMiN representation is exact, and does not involve any approximations such as would be induced through the use of the better known finite mixture of normals approach to heavy-tailed noise modelling.

2.1 Justification for use of α-stable distributions in audio applications

The above theoretical arguments in favor of α-stable distributions are supported in our application by a very good fit to real noise data. The noisy data is taken from 78rpm recordings of ethnomusicological sources carried out by Lachmann in the early 20th century. An excerpt of just over one second (44487 observations) of the recording, in which there was no musical signal present, was extracted and fitted to a stable distribution, using an approximate maximum likelihood method based on the FFT of the characteristic function[1]. Results are reported in table 1, along with the estimated parameters for a simple Gaussian and a more standard heavy-tailed Student's t model. In figure 1 we report the kernel density estimate of the dataset and the normal, the Student's t and the stable fitted densities. Although not very far from normality, the stable distribution provides a much better fit both in the central part and in the tails of the distribution with respect to both the Gaussian and the Student's t model. The estimation output of the α-stable model also highlighted a mild degree of negative asymmetry, but in order to be able to exploit the mixture of normals representation we will restrict our attention, in what follows, to the symmetric case.

Table 1: Maximum likelihood estimates of an α-stable distribution and test statistics for different null hypotheses.

	α-Stable			*Student's t*		*Gaussian*
	Est.	Std. err.		Est.		Est.
α	1.8352	0.0062	ν	5.5145		
β	-0.2226	0.0282	μ	0.1006	μ	0.0166
γ	3.6343	0.0108	σ	4.6706	σ	5.9642
δ	-0.0181	0.0173		-1.0457		

3. STATISTICAL MODELS

AR processes have been widely and successfully used in the setting of audio enhancememt [8] (and in many other areas of signal processing). Here we adopt the time-varying autoregressive (TVAR) process, in which the AR coefficients evolve over time according to certain specified dynamics. Models of this type have been employed in signal processing by, amongst others, [6], [12] and [19]. The audio signal

[1]For a more detailed description on how the maximum likelihood procedure is implemented, we refer to [16].

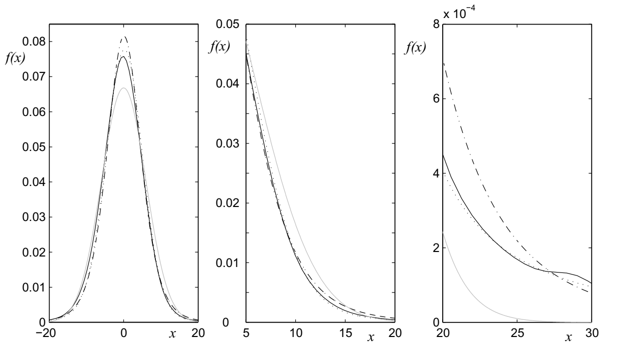

Figure 1: Kernel density (solid line), Gaussian fit (grey line), Student's t fit (dashed line) and α-stable fit (dotted line).

at time t is thus modelled as a TVAR (p) process

$$x_t = \sum_{k=1}^{p} a_{k,t} x_{t-k} + \sigma_{\epsilon_t}\epsilon_t, \qquad \epsilon_t \sim \mathcal{N}(0,1), \tag{3}$$

and is burued in symmetric α-stable noise such that the observations are

$$y_t = x_t + \gamma_{\eta_t}\eta_t, \qquad \eta_t \sim \mathcal{S}(\alpha, 0), \tag{4}$$

where σ_{ϵ_t} and γ_{η_t} represent, respectively, the standard deviation of the innovations in the true signal process and the scale of the stable noise; both are allowed to be time-varying. We furthermore assume that ϵ_t and η_t are independent. The time-varying parameter vector of the model thus has dimension $p+2$ and is given by

$$\boldsymbol{\theta}_t = (\mathbf{a}_t, \phi_{\epsilon_t}, \phi_{\eta_t}), \qquad \theta_t \in \{A_p \times \mathbb{R} \times \mathbb{R}\}$$

with

$$\mathbf{a}_t = (a_{1,t}, a_{2,t}, \ldots, a_{p,t}) \quad \phi_{\epsilon_t} = \ln \sigma^2_{\epsilon_t}, \quad \phi_{\eta_t} = \ln \gamma^2_{\eta_t};$$

where A_p is the region of stability of a stationary AR(p) process[2].

The above model can be readily expressed in state-space form. The system matrices are

$$\mathbf{A}_t = \begin{bmatrix} \mathbf{a}'_t \\ \mathbf{I}_{p-1} \ \ \mathbf{0}_{k-1\times 1} \end{bmatrix} \qquad \mathbf{B}_t = \begin{bmatrix} \sigma_{\epsilon_t} \\ \mathbf{0}_{k-1\times 1} \end{bmatrix}$$
$$\mathbf{C} = [1 \ \ \mathbf{0}_{k-1\times 1}] \qquad \mathbf{D}_t = [\gamma_{\eta_t}]$$

and, defining $\check{\mathbf{x}}_t = (x_t, x_{t-1}, \ldots, x_{t-p+1})$,

$$\check{\mathbf{x}}_t = \mathbf{A}_t\check{\mathbf{x}}_{t-1} + \mathbf{B}_t\mathbf{v}_t \qquad \mathbf{v}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{5}$$
$$\mathbf{y}_t = \mathbf{C}\check{\mathbf{x}}_t + \mathbf{D}_t\mathbf{u}_t \qquad \mathbf{u}_t \sim \mathcal{S}(\alpha, \mathbf{0}). \tag{6}$$

Now, exploiting the mixture of normal representation of a stable distribution (2), we can redefine

$$\mathbf{D}^\star_t = \left[\gamma_{\eta_t}\sqrt{\lambda_t}\right] \qquad \lambda_t \sim \mathcal{S}\left(\tfrac{\alpha}{2}, 1\right)$$

and express (6) exactly equivalently as

$$\mathbf{y}_t = \mathbf{C}\check{\mathbf{x}}_t + \mathbf{D}^\star_t\mathbf{w}_t \qquad \mathbf{w}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{7}$$

[2] This condition is only sufficient and not necessary when dealing with TVAR processes. However, regions of stability for TVAR processes are much more complex to deal with, so we have decided to enforce this simpler condition.

so that the model is expressed in conditionally Gaussian state space form. According to this approach, λ_t would be treated as a unknown parameter and incorporated into θ_t.

The evolution of θ_t over time (excluding λ_t) obeys a first order Markov process, whose parameters are assumed to be fixed and known:

$$\begin{aligned} p(\theta_0) &= p(\mathbf{a}_0)p(\phi_{\epsilon_0})p(\phi_{\eta_0})p(\lambda_0) \\ p(\theta_t|\theta_{t-1}) &= p(\mathbf{a}_t|\mathbf{a}_{t-1})p(\phi_{\epsilon_t}|\phi_{\epsilon_{t-1}})p(\phi_{\eta_t}|\phi_{\eta_{t-1}})p(\lambda_t) \end{aligned}$$

with

$$\begin{aligned} p(\phi_{\epsilon_0}) &= \mathcal{N}(0, \delta^2_{\epsilon_0}) & p(\phi_{\epsilon_t}|\phi_{\epsilon_{t-1}}) &= \mathcal{N}(\phi_{\epsilon_{t-1}}, \delta^2_{\epsilon}) \\ p(\phi_{\eta_0}) &= \mathcal{N}(0, \delta^2_{\eta_0}) & p(\phi_{\eta_t}|\phi_{\eta_{t-1}}) &= \mathcal{N}(\phi_{\eta_{t-1}}, \delta^2_{\eta}) \\ p(\lambda_t) &= \mathcal{S}(\tfrac{\alpha}{2}, 1) & p(\mathbf{a}_0) &\propto \mathcal{N}(\mathbf{0}, \Delta_{\mathbf{a}_0})\mathbb{I}_{\mathbf{a}_0\in A_p} \end{aligned} \tag{8}$$

$$p(\mathbf{a}_t|\mathbf{a}_{t-1}) \propto \mathcal{N}(\mathbf{a}_{t-1}, \Delta_{\mathbf{a}})\mathbb{I}_{\mathbf{a}_t\in A_p}.$$

4. SEQUENTIAL MONTE CARLO METHODS

We have already stated that our main goal is to reconstruct, on the basis of the observable noisy signal, the unobservable clean signal. One could be interested in simply obtaining a point estimate $\hat{x}_t$ for every time interval, but in Bayesian terms it is much more interesting to focus our analysis on the filtering distribution $p(\check{\mathbf{x}}_t, \theta_t|\mathbf{y}_{1:t})$ or on the fixed-lag smoothing distribution $p(\check{\mathbf{x}}_t, \theta_t|\mathbf{y}_{1:t+L})$, on the basis of which we can construct both point estimates and HPD intervals for x_t, for example.

Expressed in the above formulation, the model is not linear and closed-form algorithms such as the Kalman filter cannot be employed. However, it is immediately observed that, conditionally on $\theta_{0:t}$, the model is linear and Gaussian; $p(\check{\mathbf{x}}_t|\theta_{0:t}, \mathbf{y}_{1:t})$ can thus be obtained analytically using the Kalman filter.

The Kalman filter runs as follows: for $k = 1, \ldots, t$ we first set the sufficient statistics for the predictive distributions

$$\begin{aligned} \mathbf{m}_{k|k-1}(\theta_{0:k}) &= \mathbf{A}_k\mathbf{m}_{k-1|k-1} \\ \mathbf{P}_{k|k-1}(\theta_{0:k}) &= \mathbf{A}_k\mathbf{P}_{k-1|k-1}\mathbf{A}'_k + \mathbf{B}_k\mathbf{B}'_k \\ \mathbf{y}_{k|k-1}(\theta_{0:k}) &= \mathbf{C}\mathbf{m}_{k|k-1}; \end{aligned}$$

we compute

$$\mathbf{S}_k(\theta_{0:k}) = \mathbf{C}\mathbf{P}_{k|k-1}\mathbf{C}' + \mathbf{D}^\star_k\mathbf{D}^{\star\prime}_k,$$

and we finally obtain the parameters of the filtering distribution according to

$$\begin{aligned} \mathbf{m}_{k|k}(\theta_{0:k}) &= \mathbf{m}_{k|k-1} + \mathbf{P}_{k|k-1}\mathbf{C}'\mathbf{S}^{-1}_k\left(\mathbf{y}_k - \mathbf{y}_{k|k-1}\right) \\ \mathbf{P}_{k|k}(\theta_{0:k}) &= \mathbf{P}_{k|k-1} - \mathbf{P}_{k-1|k-1}\mathbf{C}'\mathbf{S}^{-1}_k\mathbf{C}\mathbf{P}_{k|k-1}. \end{aligned}$$

The filtering distribution of the state vector is thus

$$p(\check{\mathbf{x}}_k|\theta_{0:k}, \mathbf{y}_{1:k}) = \mathcal{N}\left(\mathbf{m}_{k|k}, \mathbf{P}_{k|k}\right), \tag{9}$$

and the likelihood of the last observation is

$$p(\mathbf{y}_k|\theta_{0:k}, \mathbf{y}_{1:k-1}) = \mathcal{N}\left(\mathbf{A}_k\mathbf{m}_{k|k}, \mathbf{D}^\star_k + \mathbf{A}_k\mathbf{P}_{k|k}\mathbf{A}'_k\right). \tag{10}$$

Now, since

$$p(\check{\mathbf{x}}_t, \theta_{0:t}|\mathbf{y}_{1:t}) = p(\check{\mathbf{x}}_t|\theta_{0:t}, \mathbf{y}_{1:t})p(\theta_{0:t}|\mathbf{y}_{1:t}),$$

the problem reduces to one of obtaining simulated values from $p(\theta_{0:t}|\mathbf{y}_{1:t})$ in order to produce a random sample to be used for Monte Carlo inference[3]. This is in general difficult, and an importance sampling technique can be employed. Given a probability distribution $\pi(\theta_{0:t}|\mathbf{y}_{1:t})$ which is easy to simulate from, we produce a set of M random vectors $\theta_{0:t}$ from it and assign to each one a weight

$$w_{0:t}(\theta_{0:t}) \propto \frac{p(\theta_{0:t}|\mathbf{y}_{1:t})}{\pi(\theta_{0:t}|\mathbf{y}_{1:t})}$$

to be used in Monte Carlo inference.

4.1 Particle Filters

In the above framework the data are processed in batches and, as new observations arrive, it is necessary to produce a new sample from the importance distribution (with increasingly large sample size) and reassign the importance weights. In many practical situations, however, ranging from the signal processing to the financial field, data are available on a sequential basis, and having to re-run the whole estimation as new data arrives is often not feasible when new observations arrive at a high rate.

Particle filtering methods have been recently rediscovered in independent work by [11] and [13]. The idea underlying this approach is to represent the distribution of interest by a large number of *particles* evolving over time on the basis of a simulation-based updating scheme, so that new observations are incorporated in the filter as they become available.

More formally, the objective is to update, at each time interval, $p(\theta_{0:t}|\mathbf{y}_{1:t})$ without modifying the past values of θ. The importance distribution should thus be such that

$$\pi(\theta_{0:t}|\mathbf{y}_{1:t}) = \pi(\theta_{0:t-1}|\mathbf{y}_{1:t-1})\pi(\theta_t|\theta_{0:t-1}, \mathbf{y}_{1:t}),$$

and the weights factorise as $w(\theta_{0:t}) = w(\theta_{0:t-1})w_t$ where

$$w_t \propto \frac{p(\mathbf{y}_t|\theta_{0:t}, \mathbf{y}_{1:t-1})p(\theta_t|\theta_{t-1})}{\pi(\theta_t|\theta_{0:t-1}, \mathbf{y}_{1:t})}.$$

The weights can then be updated recursively, since $w(\theta_{0:t}) = w(\theta_{0:t-1})w_t$.

It was shown in [4] that the optimal importance distribution, that is the one that minimizes the variance of the importance weights, is $p(\theta_t|\theta_{0:t-1}, \mathbf{y}_t)$. Unfortunately, this is not easy to simulate from. A simple alternative, as employed in [11, 13], is to use

$$\pi(\theta_t|\theta_{0:t-1}, \mathbf{y}_{1:t}) = p(\theta_t|\theta_{t-1}), \quad w_t \propto p(y_t|\theta_{0:t}, y_{1:t-1})$$

i.e. the importance weights are then simply proportional to the marginal likelihood. The weights are then normalized according to

$$\tilde{w}\left(\theta_{0:t}^{(i)}\right) = \frac{w\left(\theta_{0:t}^{(i)}\right)}{\sum_{j=1}^{M} w\left(\theta_{0:t}^{(j)}\right)},$$

where $\theta_{0:t}^{(i)}$ denotes the i-th *particle*. This is basically the approach employed in the seminal paper by [11], plus the resampling step which we will discuss in the following paragraph.

[3]This is an example of the Rao – Blackwellized procedure, see [4].

4.2 Resampling

It turns out that an algorithm of this kind will eventually degenerate, i.e. assign almost all the weight to a single particle. In order to overcome this problem, a resampling step is necessary. In the resampling step, particles with low importance weights are discarded and those with high importance are multiplied. More formally, after producing a set of particles from the importance distribution and having assigned to each one an appropriate weight, we associate to each particle i a number of offsprings M_i such that $\sum_{i=1}^{M} M_i = M$. After this selection step, offspring particles replace the original particles and the importance weights are reset to $1/M$, so that the set of particles can be thought of as a random sample. The resampling step can be implemented at every time interval [11], or it can be employed whenever the set of particles crosses a certain degeneracy threshold. A measure of degeneracy of the algorithm is the effective sample size [15], defined as

$$M_e = \frac{M}{1 + \mathrm{Var}(w_t)}.$$

This quantity can be estimated by

$$\hat{M}_e = \frac{1}{\sum_{i=1}^{M} \tilde{w}_t^{2(i)}};$$

when $\hat{M}_e$ drops below a certain threshold, the resampling takes place.

Several different sampling schemes have been proposed in the literature. In the following, we will denote with $\tilde{\theta}_t^{(i)}$ the i-th particle, as sampled from the importance distribution, at time t, and with $\tilde{w}_t^{(i)}$ its associated normalized importance weight. In our work, we have employed *Systematic sampling*, which has smaller variance with respect to other schemes [24]. The number of offsprings is taken to be proportional to the importance weight and is generated by simulating a set U of M uniformly distributed random variables on $[0, 1]$, taking the cumulated sum of the normalized weights

$$q_i = \sum_{j=1}^{i} \tilde{w}_t^{(j)},$$

and then setting M_i equal to the number of points in U that fall between q_{i-1} and q_i. In this case the variance is

$$\mathrm{Var}(M_i) = \bar{M}\tilde{w}_t^{\prime(i)}\left(1 - \bar{M}w_t^{\prime(i)}\right).$$

To wrap up, what the algorithm practically implements at each time interval is the following:

1. Sample M particles $\theta_t^{\dagger}$ from the importance distribution $\pi(\theta_t|\theta_{0:t-1}, \mathbf{y}_{1:t})$ and set $\theta_{0:t}^{\dagger} = (\theta_t^{\dagger}, \theta_{0:t-1})$.
2. Evaluate the importance weights according to
$$w_t \propto \frac{p(\mathbf{y}_t|\theta_{0:t}^{\dagger}, \mathbf{y}_{1:t-1})p(\theta_t^{\dagger}|\theta_{t-1})}{\pi(\theta_t^{\dagger}|\theta_{0:t-1}, \mathbf{y}_{1:t})}.$$
3. Normalize the importance weights:
$$\tilde{w}(\theta_{0:t}^{(i)}) = \frac{w\left(\theta_{0:t}^{(i)}\right)}{\sum_{j=1}^{M} w\left(\theta_{0:t}^{(j)}\right)}.$$
4. Resample if $\hat{M}_e$ below threshold by multiplying or discarding particles according to their weight to produce a new set of M particles $\theta_{0:t}$, each with weight $\tilde{w}(\theta_{0:t}) = 1/M$.

An issue which is closely related to degeneracy is that of the depletion of samples. When performing the resampling step, particles with high importance weight tend to be sampled a large number of times and it could happen that the initial set of particles ends up in collapsing into a single particle. A method to overcome this problem [14] is to sample from a kernel smoothed estimate of the target density, computed on the basis of the current set of particles. However, the drawback of this approach is that, besides raising problems concerning the choice of a specific kernel and bandwidth, it increases the Monte Carlo variance. We will examine in what follows two situations in which the depletion of samples should be seriously taken into account.

4.3 Fixed-lag smoothing

In some cases, in order to obtain a smoother estimate of the state, it is useful to consider the distribution at time t after a certain number of time intervals L. In more formal terms, instead of considering $p(\check{\mathbf{x}}_t|\mathbf{y}_{1:t})$, we focus on $p(\check{\mathbf{x}}_t|\mathbf{y}_{1:t+L})$. It is hoped that expanding the information set by the use of an appropriately chosen lag window L improves the estimates of the states. In principle, fixed-lag smoothed densities can be straightforwardly obtained by the general algorithm proposed above, by simply extracting signal estimates at time $t-L+1$ from the particles. However, it turns out that such a scheme, where states at time $t-L+1$ could have been resampled up to L times, leads to serious degeneracy as L grows larger.

In order to overcome this degeneracy problem, an MCMC approach similar to [24] can be adopted. Let us consider that, at time $t+L$, the particles are distributed according to $p(\theta_{0:t+L}|\mathbf{y}_{1:t+L})$; the idea is to apply to each particle a Markov transition kernel K^* with invariance distribution $p(\theta_{0:t+L}, \mathbf{y}_{1:t+L})$ in order to introduce diversity among the particles.

If we denote with $\theta'^{(i)}_{0:t+L}$ the i-th particle after the resampling stage, the MCMC proceeds by sampling each particle and state $\theta^{(i)}_k$ according to target density $p(\theta_k|\theta^{(i)}_{-k}, \mathbf{y}_{1:t+L})$, where

$$\theta^{(i)}_{-k} = \left(\theta'^{(i)}_{0:t-1}, \theta^{(i)}_t, \ldots, \theta^{(i)}_{k-1}, \theta'^{(i)}_{k+1}, \ldots, \theta'^{(i)}_{t+L}\right),$$

and $k = t, t+1, \ldots, t+L$. A Metropolis-within Gibbs sampling MCMC scheme is used to achieve this and full details, including the forward-backward Kalman filter for efficient implementation, may be found in[24].

4.4 Static parameters

The degeneracy problem is however much more severe whenever the particle filter has to deal with the estimation of static parameters. The prior $p(\theta_{t+1}|\theta_t)$ would have probability mass 1 at θ_t, so the particles are never updated and rejuvenated and they eventually collapse on a few – and sometimes even one – single value. In our specific case, we have up to now assumed the stable law tail parameter α to be known. This is rarely the case in practical applications. Furthermore, whereas in static estimation problems one could somehow pre-estimate static parameters, in our sequential estimation case this is obviously impossible. Fixing the static parameters to arbitrarily chosen guesses is in general very bad practice. In our specific case, however, a few experiments, not reported here for sake of brevity, have reported that, even if the guessed α is not very close to the actual one, the results are still satisfactory and the improvement in the SNR remains at the same order of magnitude. However, it would surely be preferable to estimate α together with the other parameters as the data are processed.

Several approaches to this problem are available. The MCMC schemes above for fixed lag smoothing can be adapted to the static parameter case, for example, although in our case this led to a filter of ever growing computational complexity and so was not adopted. Another approach to overcome the degeneracy problem is to introduce artificial parameter evolution, that is, simply pretending that static parameters are indeed time-varying by adding a noise term at each time interval. The problem is that in doing so we introduce additional variability by "throwing away" information. [14] propose a method for quantifying this loss of information and devise an artificial parameter evolution scheme immune to this problem. To focus the attention on our specific case, we note that the static parameter is the stability index α. Introducing artificial parameter evolution is equivalent to consider a model in which α is replaced by its time-varying analog α_t which evolves according to

$$\alpha_t = \alpha_{t-1} + \zeta_t, \qquad \zeta_t \sim \mathcal{N}(0, \omega_t).$$

In a situation in which α is fixed, the posterior distribution $p(\alpha|\mathbf{y}_{1:t})$ could be characterized by its Monte Carlo mean and variance $\bar{\alpha}_t$ and s_t^2. It is immediate to observe that, in the case of artificial parameter evolution, the Monte Carlo variance increases to $s_t^2 + \omega_t$. The Monte Carlo approximation can be expressed as kernel smoothed density of the particles as

$$p(\alpha|\mathbf{y}_{1:t}) \approx \sum_{j=1}^{M} \kappa_t^{(j)} \mathcal{N}\left(\alpha_{t+1}|\alpha_t^{(j)}, \omega_t\right).$$

Now the target variance s_t^2 can be expressed as

$$s_t^2 = s_{t-1}^2 + \omega_t + 2\mathrm{Cov}(\alpha_{t-1}, \zeta_t),$$

so if we choose

$$\mathrm{Cov}(\alpha_{t-1}, \zeta_t) = -\frac{\omega_t}{2}$$

we have managed to avoid the loss of information. A simple particular case in which this can be achieved is to consider

$$\omega_t = s_t^2 \left(\frac{1}{\delta} - 1\right),$$

where δ is a discount factor in $(0, 1]$; the authors suggest its value to be chosen around 0.95-0.99. If we define $d = \frac{3\delta-1}{2\delta}$, the conditional density evolution becomes

$$p(\alpha_{t+1}|\alpha_t) \sim \mathcal{N}\left(\alpha_{t+1}|d\alpha_t + (1-d)\bar{\alpha}_t, h^2 s_t^2\right), \qquad (11)$$

where

$$h^2 = 1 - d^2 = 1 - \left(\frac{3\delta - 1}{2\delta}\right)^2,$$

so that sampling from (11) is equivalent to sampling from a kernel smoothed density in which the smoothing parameter h is controlled via the discount factor δ.

5. EXPERIMENTS AND RESULTS

In this section we will show how the sequential Monte Carlo method outlined above performs on both simulated and real audio data. As a benchmark of model performance, we will use the signal to noise ratio, defined as

$$\mathrm{SNR} = 10 \log_{10} \frac{\sum_{t=1}^{T} x_t^2}{\sum_{t=1}^{T} (x_t - z_t)^2},$$

where x_t is the clean signal and z_t represents, in turn, the observed noisy signal and the filtered state. We will start by considering the simplest case, that is the one in which α is

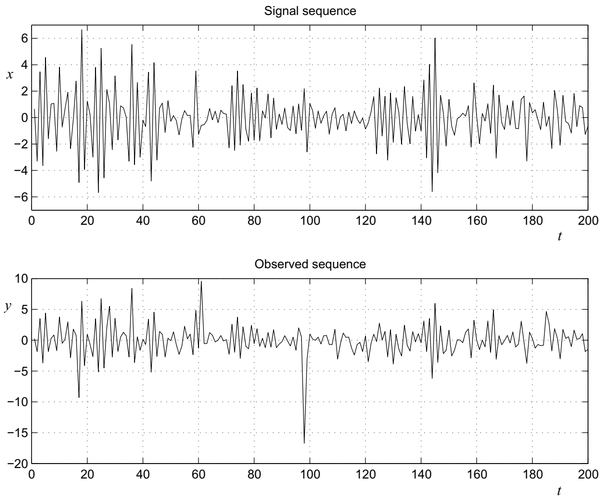

Figure 2: Clean and noisy signal, synthetic data.

known a priori and we do not perform fixed-lag smoothing, so that there is no need for the MCMC step outlined in the above subsection. The importance function was taken to be the prior $p(\theta_t|\theta_{t-1})$; as a resampling scheme, we will use systematic sampling, applied at each time step.

We have generated a synthetic signal of 200 observations with parameters $\Delta_{\mathbf{a}_0} = 2\mathbf{I}$, $\Delta_{\mathbf{a}} = 0.0005\mathbf{I}$, $\delta^2_{\epsilon_0} = 0.2$, $\delta^2_\epsilon = 0.005$, $\delta^2_{\eta_0} = 0.5$, $\delta^2_\eta = 0.00005$; the signal was then corrupted with symmetric α-stable noise with $\alpha = 1.4$. The SNR of the noisy observations was 0.83dB. The synthetic data are depicted in figure 2.

Using a simple Gaussian model, as the one proposed by [24], obviously leads to poor results. Especially when the signal is highly corrupted by the noise peaks, the filtered states are very near to the observations, according to the low likelihood of such extreme values under the Gaussian noise assumption. Furthermore the extreme observations are somehow "absorbed" by jumps in the variance of the signal. The overall improvement in SNR was of 0.86dB, with RMSE 1.6947.

On the other hand, the use of the stable model greatly reduces the influence of the extreme noise observations, achieving a SNR improvement of 5.12dB with RMSE 1.0382; in particular, we note that the filter is not misled by extreme observations as it happened in the Gaussian case. Similar results hold when α is estimated along with the other parameters. The prior we used for α was a simple uniform distribution on $[0.2, 2]$[4], and we fixed the discount factor δ in (11) to 0.95. The evolution of the stability index is depicted in the top graph of figure 4 along with the 95% quantile bands. The SNR improvement is 5.13dB with RMSE 1.0372, nearly identical to the case analyzed earlier in which we fixed α to its true value. The evolution of the kernel smoothed posterior distribution of α in for the last intervals is presented in figure 5.

In order to get insights about the appropriate number of particles to be used, we have performed a Monte Carlo experiment consisting of 50 independent replications. All experiments were performed on a laptop computer with a 2.66GHz Intel® Pentium® IV processor with 512Mb RAM. The results, reported in Table 2 seem to indicate that using more than 300 particles does not lead to a significantly improved performance despite the increase in computational effort. A number of particles between 100 and 300 seems to be a good compromise between speed and accuracy.

[4] Values of α smaller than 0.2 were ruled out in order to avoid overflows.

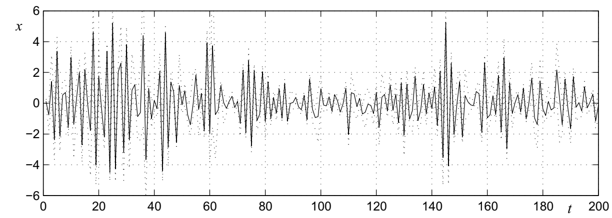

Figure 3: Filtered signal (solid line) with 95% quantile bands (dotted lines), stable noise, $\alpha = 1.4$.

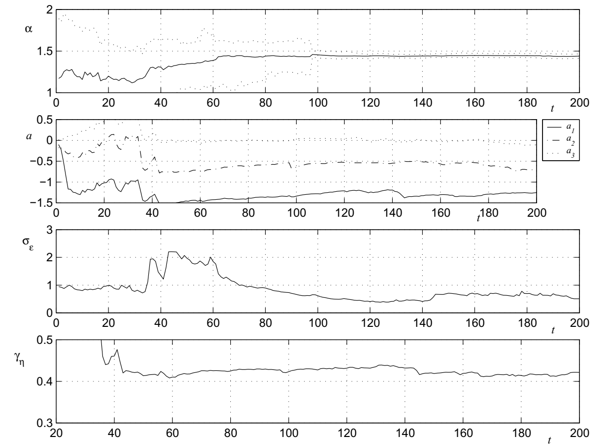

Figure 4: Estimated parameters of the model, stable noise.

Table 2: RMSE and mean and standard deviation (in parentheses) of SNR improvement, over 50 independent replications, for different number of particles M. The last row reports the average time (in seconds) required to process one observation.

M	10	50	100	300	500
RMSE	1.3688	0.9921	0.9893	0.9892	0.9892
SNR	2.8651	4.8815	5.1515	5.4857	5.4786
	(1.6433)	(0.7991)	(0.5344)	(0.2708)	(0.2222)
Time	0.0219	0.0897	0.1432	0.3810	0.6407

Concerning the fixed-lag smoothing, we have performed a simulation experiment consisting of 50 independent replications over 100 particles for different lengths of the lag window. Results are reported in Table 3 and suggest that an optimal lag window could be between 5 and 10. The last simulation experiment we have performed consisted in artificially corrupting with symmetric α-stable noise a clean audio source; we have used the first 6.75 seconds of the Boards of Canada's "Music is Math" from the album "Geogaddi", ripped in PCM format (44.1KHz, 16 bit, mono) from the original CD. This audio source was produced on computer, so it presents no kind of corruption or background noise. The parameters of the artificial noise were set to $\alpha = 1.7$, $\delta = 0$, and the scale parameter γ was evolved from its initial value 0.01 according to a Markov process as in (8), with $\delta = 0.01$. The resulting SNR is 3.9564. For illustration

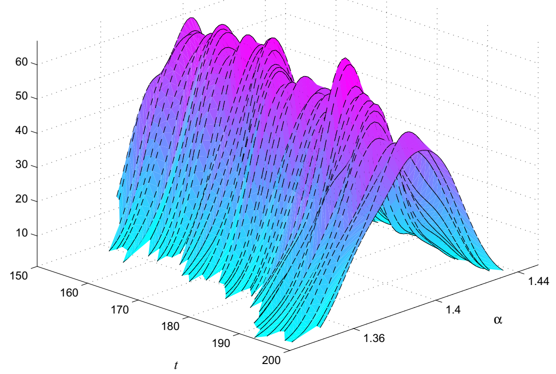

Figure 5: Kernel smoothed posterior densities of α in for $t = 150, \ldots, 200$.

Table 3: RMSE and mean and standard deviation (in parentheses) of SNR improvement, over 50 independent replications, for different length of the lag window L with 100 particles. The last row reports the average time (in seconds) required to process one observation.

	$L=0$	$L=5$	$L=10$	$L=20$
RMSE	0.9892	1.0151	0.8893	0.9561
SNR	5.1515	6.1061	6.1907	5.8457
	(0.5344)	(0.4889)	(0.5483)	(0.6312)
Time	0.1432	1.0734	1.9126	3.5102

purposes, the filter was first run on an excerpt of 1000 observations (200001 to 201000 out of 261072, input SNR 8.72), clean, noisy and filtered signal for this excerpt are displayed in figure 6. We have employed 200 particles, with a fixed-lag smoothing window of length 5; the filter performed remarkably well, achieving a SNR improvement of 8.5308.

The same filter was then applied to the whole series, yielding again a remarkable SNR improvement. The reconstructed audio source was then recoded in audio format, and informal listening tests confirmed the reduction of the noise. In particular, the filter performed very well in removing the peaks but left a small amount background noise[5], sounding like a feeble "hiss".

A very sensitive issue we had to deal with was that of the choice of the prior for the scale parameters σ and γ. Preliminary experiments pointed out how values very far from their true counterparts can lead to very poor performance, mainly owing to the trade-off effect between the scale and the tail-thickness parameter and the slow evolution speed of the scale parameter. For example, a too small value of γ can lead α to decrease to compensate the effect. In the present case, we have bypassed the problem by using a strongly informative normal prior centered on the true value of γ, which was obviously known a priori. The issue deserves however further attention, especially because when we are interested to process the observations on-line, unless we decide to discard the very first observations as if it was a sort of burn-in period, it is not even possible to pre-estimate the scale parameter in order to get insights on an appropriate prior from which to sample the initial particles. It would certainly be preferable to specify an appropriate joint prior that takes

[5]All audio examples presented in this paper can be downloaded at the URL *http://www.ds.unifi.it/mjl/sound.htm*.

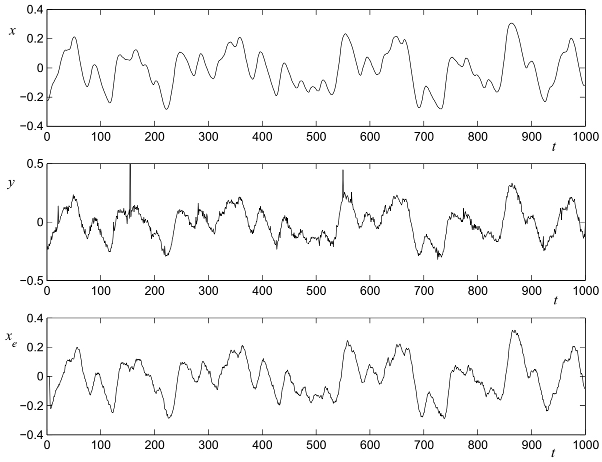

Figure 6: Excerpt of clean, noisy and reconstructed signal for Boards of Canada's "Music is Math".

into account the dependence structure between the two parameters. This issue however deserves further attention and will be the subject of future research.

We are now in position to consider an application of the above methodology to genuine corrupted audio data from the Lachmann database. As we have anticipated, the audio source we will refer to if a set of recordings of songs taken in Palestine in the early twentieth century; this audio source is extremely noisy and corrupted; as we have pointed out in section 2, the noise of this audio source is modelled very well by an α-stable distribution. We have applied the particle filter, with $M = 100$ and $L = 100$ to a short an excerpt of two seconds of one of the audio tracks. Given the large number of lags involved, we have decided not to perform the MCMC step in order to reduce the computational time. The results were encouraging: the peaks in the corrupted audio signal were removed, leaving behind a "seemingly white" background noise that could be dealt with by traditional filtering methods. The final estimated value for α was 1.5509.

6. CONCLUSIONS

We have proposed and tested methods for performing online Bayesian filtering in TVAR models with symmetric α-stable noise distribution. Using such a of distribution allows for more flexibility and permits successful modelling of the heavy-tailed noise which is often observed in empirical audio time series [8]. The performance of this filtering method was assessed on both simulated and real data, and the analysis of a genuinely degraded audio source suggested that α-stable distributions are particularly well suited to model this kind of noise.

The reason for which we considered only symmetric cases of α-stable distributions instead of the more general asymmetric version is that they can be represented exactly as a scale mixtures of normals. This useful property, that allows us to use the Kalman filter by expressing the model in conditionally Gaussian form, does not hold for the more general asymmetric case. In the general case, one should resort to more standard techniques to obtain the likelihood of every particle, but the necessity to perform the inversion of the characteristic function via the FFT at each time interval and, within a given time interval, for each particle, would lead to excessive computational requirements, at least according to the power of the machines available to us. In fact we believe from observation that the α-stable distributions involved in audio noise are very close to symmetric, so we do not regard this restriction as a serious limit to the methods in practice.

Although we have focused our analysis on symmetric α-stable distributions, this approach has much more generality and can routinely be extended to other situations in which the distribution of the noise can be represented as a scale mixture of normals; it is in fact sufficient to modify the distribution of the scaling factor λ. Distributions that can be expressed as scale mixtures of normals include the logistic, Student's t and power exponential [25]. In particular, the Student's t and the power exponential distribution are especially appreciated in the setting of noise modelling and we will present here for reference the densities that should be employed for the scale factor λ.

If the noise has t distribution with ν degrees of freedom, scale parameter σ and location parameter μ

$$\epsilon_i \sim t(\nu, \mu, \sigma),$$

the scaling factor has inverse gamma distribution with shape parameter $\nu - \frac{1}{2}$ and scale parameter 2 [1]:

$$\epsilon_i = \delta + \gamma\sqrt{\lambda_i}u_i, \qquad \lambda_i \sim \mathcal{I}g\left(\nu - \tfrac{1}{2}, 2\right), \; u_i \sim \mathcal{N}(0,1).$$

The (standardized) power exponential distribution, sometimes referred to as generalized error distribution (GED), has probability density function

$$f(x) \propto \exp\left(|x|^{-\alpha}\right),$$

with $\alpha \in [1, 2]$; the case $\alpha = 2$ obviously corresponds to a Gaussian distribution, and $\alpha = 1$ to a Laplace, or double exponential, distribution. If ϵ_i has power exponential distribution with parameters α, μ and σ, the scaling factor can be shown [25] to have density

$$p(\lambda_i) \propto \lambda_i^{-2} s\left(\lambda_i^{-2}; \tfrac{\alpha}{2}, 1\right),$$

where $s(\cdot; \alpha, \beta)$ denotes the probability density function of a standard stable distribution with tail parameter α and asymmetry parameter β. Although this density cannot be expressed in closed form, simulated values can be readily obtained using the approach of [3].

In general, t distributions and power exponentials are far more popular than the α-stable for heavy tailed modelling purposes; in our opinion this is mainly because of their simplicity. However, as we have observed in section 2, the α-stable distribution fits our data much better than the Student's t. Moreover, in our framework the α-stable and GED models will involve approximately the same computational burden as that for the (apparently simpler) Student's t case, since the generation of stable law random numbers takes roughly the same magnitude of computation as that needed to produce inverse gamma distributed random numbers.

To conclude, we have presented practical Monte Carlo methods for on-line estimation of TVAR models in the presence of α-stable noise. The methods are accurately able to infer the signal state as well as unknown parameters, including the challenging α parameter of the stable distribution. Results so far are promising for some of the most demanding degraded audio sources obtained from early ethnomusicological archives.

REFERENCES

[1] D.F. Andrews and C.L. Mallows, "Scale mixtures of normal distributions", *Journal of the Royal Statistical Society B*, vol. 36 pp. 99–102, 1974.

[2] D.J. Buckle, "Bayesian inference for stable distributions", *Journal of the American Statistical Association*, vol. 90 pp. 605–613, 1995.

[3] J.M. Chambers, C. Mallows, and B.W. Stuck, "A method for simulating stable random variables", *Journal of the American Statistical Association*, vol. 71 pp. 340–344, 1976.

[4] A. Doucet, S.J. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering", *Statistics and Computing*, vol. 10 pp. 197–208, 2000.

[5] A. Doucet, S.J. Godsill, and M. West, "Monte Carlo filtering and smoothing with application to time-varying spectral estimation", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2000.

[6] W. Fong, A. Doucet, S.J. Godsill, and M. West, "Monte Carlo smoothing with application to speech enhancement", *IEEE Transactions on Signal Processing*, 50:438–449, 2002.

[7] B.V. Gnedenko and A.N. Kolmogorov, *Limit Distributions for Sums of Independent Random Variables.* Reading: Addison-Wesley, 1954.

[8] S. J. Godsill and P. J. W. Rayner, "Digital Audio Restoration: A Statistical Model-Based Approach", Berlin: Springer, ISBN 3 540 76222 1, 1998[6]

[9] S.J. Godsill, "MCMC and EM-based methods for inference in heavy-tailed processes with α-stable innovations", In *Proceedings of the IEEE Signal Processing Workshop on Higher-order Statistics*, 1999.

[10] S.J. Godsill, A. Doucet, and M. West, "Monte Carlo smoothing for non-linear time series", *Journal of the American Statistical Association*, in press, 2004.

[11] N.J. Gordon, D.J. Salmond, and A.F.M. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation", *IEE Proceedings-F*, vol. 140 pp. 107–113, 1993.

[12] P. Ha, and S. Ann, "Robust time-varying prametric modelling of voiced speech", *Signal Processing*, vol. 42 pp. 311–317, 1995.

[13] G. Kitagawa, "Sequential Monte Carlo filter and smoother for non-Gaussian nonlinear state space models", *Journal of Computational and Graphical Statistics*, vol. 5 pp. 1–25, 1996.

[14] J. Liu, and M. West, "Combined parameter and state estimation in simulation-based filtering", In A. Doucet, J.F.G. de Freitas, and N.J. Gordon, editors, *Sequential Monte Carlo Methods in Practice. New York.* Springer-Verlag, New York, 2001.

[15] J.S. Liu, and R. Chen, Sequential Monte Carlo methods for dynamic systems. *Journal of the American Statistical Association*, vol. 93 pp. 1032–1044, 1998.

[16] M.J. Lombardi, "Bayesian inference for α-stable distributions: A random walk MCMC approach", Dipartimento di Statistica "G. Parenti", Universit degli studi di Firenze, 2004.

[17] J.H. McCulloch, Simple consistent estimators of stable distribution parameters. *Communications in Statistics: Simulation and Computation*, vol. 15 pp. 1109–1136, 1986.

[18] S. Mittnik, S.T. Rachev, T. Doganoglu, and D. Chenyao, "Maximum likelihood estimation of stable Paretian models", *Mathematical and Computer Modelling*, vol. 29 pp. 275–293, 1999.

[19] M. Niedźwiecki and K. Cisowski, "Adaptive scheme for elimination of broadband noise and impulsive disturbances from AR and ARMA signals", *IEEE Transactions on Signal Processing*, vol. 44 pp. 528–537, 1996.

[20] C. L. Nikias and M. Shao, "Signal Processing with α-Stable Distributions and Applications", John Wiley and Sons, 1995.

[21] J.P. Nolan, "Numerical computation of stable densities and distribution functions", *Communications in Statistics: Stochastic Models*, vol. 13 pp. 759–774, 1997.

[22] G. Samorodnitsky and M.S. Taqqu, "Stable non-Gaussian random processes: stochastic models with infinite variance", New York: Chapman and Hall, 1994

[23] E.G. Tsionas, "Monte Carlo inference in econometric models with symmetric stable distributions", *Journal of Econometrics*, vol. 88 pp. 365–401, 1999.

[24] J. Vermaak, C. Andrieu, A. Doucet, and S.J. Godsill, "Particle methods for Bayesian modelling and enhancement of speech signals", *IEEE Transactions on Speech and Audio Processing*, vol. 10 pp. 173–185, 2002.

[25] M. West, "On scale mixtures of normal distributions", *Biometrika*, vol. 74 pp. 646–648, 1987.

[6] see `http://www-sigproc.eng.cam.ac.uk/sjg/springer/`